AC–_____
Item No. _ _____

UNIVERSITY OF MUMBAI

Syllabus for F.Y.B.Sc.
Programme: B.Sc.
Subject : Information
Technology
Semester – I and II

(CBCS)
(Choice Based Credit System with effect from the
academic year 2022-2023)
(To introduce with effect from the academic year
2022-2023)

UNIVERSITY OF MUMBAI

Syllabus for Approval

Sr. No.	Heading	Particulars
1	Title of the Programme	**F.Y.B.Sc. Sem. I & II (Information Technology)**
2	Eligibility for Admission	Ordinance no. O.5051 Circular no. UG/284 of 2007 dated 16th June 2007
3	Passing Marks	40%
4	Ordinances / Regulations (if any)	As applicable for all B.Sc. Courses
5	No. of Years / Semesters	Three years – Six Semesters
6	Level	~~P.G.~~ / U.G./ ~~Diploma~~ / ~~Certificate~~ **(Strike out which is not applicable)**
7	Pattern	~~Yearly~~ / Semester **(Strike out which is not applicable)**
8	Status	Revised / ~~New~~ / ~~Amended~~ **(Strike out which is not applicable)**
9	To be implemented from Academic Year	From Academic Year **2022-2023**

Signature
Chairman Name Dr. R. Srivaramangai
BOS Chairman in Information Technology

Dr. Anuradha Majumdar
Dean, Science and Technology

PREAMBLE

The B.Sc. Information Technology programme was started in 2001 with an aim to make the students employable and impart industry oriented training. The main objectives of the course are:

- To think analytically, creatively and critically in developing robust, extensible and highly maintainable technological solutions to simple and complex problems.
- To apply their knowledge and skills to be employed and excel in IT professional careers and/or to continue their education in IT and/or related post graduate programmes.
- To be capable of managing complex IT projects with consideration of the human, financial and environmental factors.
- To work effectively as a part of a team to achieve a common stated goal.
- To adhere to the highest standards of ethics, including relevant industry and organizational codes of conduct.
- To communicate effectively with a range of audiences both technical and non-technical.
- To develop an aptitude to engage in continuing professional development.

The new syllabus is aimed to achieve the objectives. The syllabus spanning three years covers the industry relevant courses. The students will be ready for the jobs available in different fields like:

- Software Development (Programming)
- Website Development
- Mobile app development
- Internet of Things
- Software Testing
- Networking
- Database Administration
- System Administration
- Cyber Law Consultant
- GIS (Geographic Information Systems)
- IT Service Desk
- Security
- Technical communication skills
- Green IT

  And many others

Name of Dean (Dean, Faculty of Science and Technology) : Dr. Anuradha Majumdar

Name of Associate Dean (Associate Dean, Faculty of Science and Technology) Prof. Shivram Garje

   Name of Chairperson (BoS) : Dr. Mrs. R. Srivaramangai

   Member(BoS) : Dr. Hiren Dand

   Member(BoS) : Dr. Abhijeet Kale

   Member(BoS) : Dr. Santosh Singh

   Member(BoS) : Dr. Rajendra Patil

   Member(BoS) : Dr. Mandar Bhave

Semester 1			
Course Code	**Course Type**	**Course Title**	**Credits**
USIT101	Core Subject	Programming Principles with C	2
USIT102	Core Subject	Digital Logic and Applications	2
USIT103	Core Subject	Fundamentals of Database Management Systems	2
USIT104	Core Subject	Computational Logic and Discrete Structure	2
USIT105	Ability Enhancement Skill Course	Technical Communication Skills	2
USIT1P1	Core Subject Practical	Programming Principles with C Practical	2
USIT1P2	Core Subject Practical	Digital Logic and applications Practical	2
USIT1P3	Core Subject Practical	Fundamentals of Database Management Systems Practical	2
USIT1P4	Core Subject Practical	Computational Logic and Discrete structure Practical	2
USIT1P5	Ability Enhancement Skill Course Practical	Technical Communication Skills Practical	2
		Total Credits	**20**

Semester 2			
Course Code	**Course Type**	**Course Title**	**Credits**
USIT201	Core Subject	Object Oriented Programming with C++	2
USIT202	Core Subject	Fundamentals of Micro Processor and Microcontrollers	2
USIT203	Core Subject	Web Applications Development	2
USIT204	Core Subject	Numerical Methods	2
USIT205	Ability Enhancement Skill Course	Green IT	2
USIT2P1	Core Subject Practical	Object Oriented Programming with C++ Practical	2
USIT2P2	Core Subject Practical	Fundamentals of Micro Processor and Microcontrollers Practical	2
USIT2P3	Core Subject Practical	Web Applications Development Practical	2
USIT2P4	Core Subject Practical	Numerical Methods Practical	2
USIT2P5	Ability Enhancement Skill Course Practical	PL/SQL Practical	2
		Total Credits	**20**

SEMESTER I

B. Sc (Information Technology)		Semester – I		
Course Name: Programming Principles with C		Course Code: USIT101		
Periods per week (1 Period is 50 minutes)		5		
Credits		2		
			Hours	Marks
Evaluation System	Theory Examination		2	75
	Internal		--	25

Course Objectives: 1. To develop the logical ability of the student.
2. Basic concepts to be cleared using suitable examples.
3. Different approach towards the problem.
4. To handle the errors and find suitable solution.
5. Debugging the code.

Unit	Details	Lectures
I	Introduction: Algorithms, History of C, Structure of C Program. Program Characteristics, Compiler, Linker and preprocessor, pseudo code statements and flowchart symbols, Desirable program characteristics. Program structure. Compilation and Execution of a Program, C Character Set, identifiers and keywords, data types and sizes , constants and its types, variables, Character and character strings, typedef, typecasting	12
II	Type of operators: Arithmetic operators, relational and logical operators, Increment and Decrement operators, assignment operators, the conditional operator, Assignment operators and expression, Precedence and order of Evaluation Block Structure, Initialization, C Preprocessor Control Flow: Statements and Blocks, If-Else, Else-If, Switch, Loops- While and For Loops-Do-while, Break and Continue, Goto and Labels	12
III	Functions and Program Structure: Basics of functions. User defined and Library functions, Function parameters, Return values, Recursion External variables, Scope Rules, Standard Input and Output, Formatted Output-printf() and Formatted Input- scanf(), Line Input and Output, Error Handling- StdErr and Exit, Header Files	12
IV	Pointer and Arrays Pointer and Addresses, Pointer and Function Arguments, Pointer and Arrays, Address Arithmetic, Character Pointers and Functions, Pointer Arrays: Pointers and Functions, Multidimensional Array, Command-line Arguments, Pointers to Functions, Dynamic memory allocation	12
V	Structures: Basics of structures, Structures and Functions, Arrays of Structures, Pointers to Structures, Unions, Bit-fields, File management in C: Defining and Opening file, Closing a file, Input / Output operations on file, Error handling in C, Random access to files, Command line arguments.	12

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Programming Language	Brian W. Kernighan and Denis M. Ritchie.	PHI	2nd	1988
2.	Mastering C	K R Venugopal	Tata McGraw-Hill	6th	2007
3.	Programming with C	Byron Gottfried	Tata McGRAW-Hill	2nd	1996
4.	Let us C	Yashwant P. Kanetkar	BPB publication		
5.	Programming in ANSI C	E.Balagurusamy	Tata McGraw-Hill	7th	1982

Course Outcomes:

Learners will be able to,

1. **Learn the basic principles of programming.**
2. **Develop of logic using algorithm and flowchart.**
3. **Acquire the information about data types.**
4. **Understanding of input and output functions.**
5. **Enhance advanced concepts using program.**

B. Sc (Information Technology)		Semester – I	
Course Name: Programming Principles with C Practical		Course Code: USIT1P1	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2½	50
	Internal	--	--

Course Objectives:
1.   **To develop the logic of the student.**
2.   **Describe loops and decision making using programs.**
3.   **Practical use of operators.**
4.   **Illustration of the difficult concepts using programming examples.**
5.   **Discussion of the relevant concepts using program.**

List of Practical:	
1.	a. Write an algorithm and draw flowchart for Area of circle. b. Write an algorithm and draw flowchart to print the given no. is even or odd. c. Write an algorithm and draw flowchart to print 1 to 10 numbers. d. Write an algorithm and draw flowchart for sum of 1 to 5 numbers. e. Write an algorithm and draw flowchart to compute the addition of digits of a given number.
2.	a. Write a program using while loop to reverse the digits of a number. b. Write a program to calculate the factorial of a given number. c. Write a program to find the roots of quadratic equation. d. Write a program to print the Fibonacci series.
3.	a. Write a program in C to check entered character vowel or consonant b. Write a program to C program to print day name of week using switch-case. c. Write a program to read three values from keyboard and print out the largest of them without using if statement.
4.	a. Write a program to print the pattern of asterisks as shown below : * * * * * * * * * * b. Write a program to print the pattern of asterisks as shown below : * * * * * * * * * * * * * * * **c. Write a program to print Floyd's Triangle.**
5.	a. Write a program to print area of square using function.

	b. Write a program using recursive function. **c. Write a program to square root, abs() value using function.** **d. Write a program using goto statement.**
6.	a. Write a program to print rollno and names of 10 students using array. b. Write a program to read a matrix of size m*n. **c. Write a program to sort the elements of array** in ascending or descending order.
7.	a. Write a program to extract the portion of a character string and print the extracted part. b. Write a program to find the given string is palindrome or not. **c. Write a program to using strlen(), strcmp() function.**
8.	a. Write a program to display the values using different data types and its address using pointer. b. Write a program to perform addition and subtraction using pointer.
9.	a. Write a program to copy the contents of the file from one file into other. b. Write a program to print the structure using • Title • Author • Subject • Book ID Print the details of two students.
10.	a. Create a mini project on "Bank management system" . The program should be menu driven.

Course Outcomes:

Learners will be able to,

1. **Develop applications.**
2. **Work with textual information, characters and strings.**
3. **Understand of a functional hierarchical code organization**
4. **Debug the program**
5. **Understand the differences between syntax errors, runtime errors, and logic errors.**

B. Sc (Information Technology)		Semester – I		
Course Name: Digital Logic and Applications		Course Code: USIT102		
Periods per week (1 Period is 50 minutes)		5		
Credits		2		
			Hours	Marks
Evaluation System	Theory Examination		2	75
	Internal		--	25

Course Objectives:
1. To introduce the basics of logic in digital electronics as an entry level course.
2. To interpret and assess number systems and the conversions of number systems
3. To analyze the boolean expressions and reduce the expression to the minimum.
4. To design simple logic circuits using tools such as Boolean Algebra and Karnaugh Mapping.
5. To understand the state of a memory cell and its types using flip-flops.
6. To create simple digital systems using counters, registers etc.

Unit	Details	Lectures
I	**Digital Systems and Binary numbers** Introduction to Number systems, Positional Number systems, Conversions (converting between bases), Non positional number systems, Unsigned and Signed binary numbers, Binary Codes, Number representation and storage in computer system. **Logic gates and Logic Circuits** Basic and Universal Gates	12
II	**Boolean algebra and Gate level minimization** Introduction, Postulates of Boolean Algebra, Two Valued Boolean Algebra, Principle of Duality, Basic Theorems of Boolean Algebra, Boolean Functions and their Representation, Gate-Level Minimization (Simplification of Boolean Function), Quine-McCluskey Method, Review questions	12
III	**Combinational logic** Introduction, Analysis and Design Procedure for Combinational Logic Circuits, Types of Combinational Circuit, Review Questions	12
IV	**Sequential circuits** Introduction, Latch, Flip-Flops, Registers, Counters, Review Questions	12
V	**Applications** Bit Arithmetic and Logic unit, Carry lookahead generator, Binary Multiplication and Division algorithm, Booth's multiplication algorithm	12

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Digital Logic Design	Sonali Singh	BPB publications	1st	2015
2.	Fundamentals of Digital Electronics and Logic Design	Subir Kumar Sarkar, Asish Kumar De, Souvil Sarkar	Pan Stanford Publishing	1st	2014
3.	Digital Electronics Principles, Design and Applications	Anil K Maini	Wiley	1st	2007
4.	Fundamentals of Logic	Charles H Roth, Jr. , Larry L	Cengage	7th	2014

	Design	Kinney	Learning		
5.	Digital Principles and Applications	Donald P Leach Albert Malvino Goutam Saha	TMH	8th	2015

Course Outcomes:

Learners will be able to,

1. Apply number conversion techniques in real digital systems
2. Solve boolean algebra expressions
3. Derive and design logic circuits by applying minimization in SOP and POS forms
4. Design and develop Combinational and Sequential circuits
5. Understand and develop digital applications

B. Sc (Information Technology)			Semester – I	
Course Name: Digital Logic and Applications Practical			Course Code: USIT1P2	
Periods per week (1 Period is 50 minutes)			3	
Credits			2	
			Hours	Marks
Evaluation System		Practical Examination	2½	50
		Internal	--	--

Course Objectives:
1. To apply and test the gates learnt using various IC's .
2. To evaluate the Boolean expression to reduce and minimize the gates used

1.	Study of basic gates and Universal gates
a.	To verify the truth tables of OR, AND, NOR, NAND, EX-OR, EX-NOR gates
b.	To study IC 7400, 7402, 7404, 7408, 7432, 7486, 74266
c.	To implement and verify NAND and NOR as Universal gates
2.	Study of Boolean expressions
a.	To verify De Morgan's laws
b.	Implement the given expression using a minimum number of gates.
c.	Implement the given expression using a minimum number of ICs.
3.	Design of Combinational Circuits using K-maps
a.	Design and implement combinational circuits for the given problem/problems using minimization techniques of K-maps.
4.	Design and implement code converters
a.	Design the circuit and implement Binary to gray code converter
b.	Design the circuit and implement Gray to Binary code converter
c.	Design the circuit and implement Binary to BCD code converter
d.	Design the circuit and implement Binary to XS-3 code converter
5.	Implement Adder and Subtractor circuits
a.	Design the circuit and implement Half Adder and Full Adder
b.	Design the circuit and implement BCD Adder, XS-3 Adder , Binary Subtractor

6.	**Design and implement Arithmetic circuits**
a.	Design and implement 2-by-2 bit multiplier
7.	**Implement Encoders and Decoders**
a.	Design and implement 8: 3 encoder
b.	Design and implement 3:8 decoder
8.	**Multiplexers and Demultiplexers**
a.	Design and Implement 4:1 multiplexer
b.	Design and Implement 1:4 demultiplexer
c.	Study IC 74151 8: 1 multiplexer and implement the expression
d.	Study IC 74138 3: 8 decoder and implement the expression
9.	**Study of Flipflops and Counters**
a.	Study of IC's 7473, 7474, and 7476
b.	Design a 3-bit ripple/ synchronous counter using IC 7473 and required gates
10.	**Design of Shift Registers**
a.	Design of Shift registers using IC 7474
b.	Implementation of digits using seven segment displays

Course Outcomes:

Learners will be able to,

1. **Construct basic and universal logic circuits.**
2. **Verify the functionalities of various IC's.**
3. **Design circuits using K-maps minimization technique**
4. **Design and test Encoders, Decoders, Multiplexers and Demultiplexers**
5. **Design and develop logic for Registers, Counters and its applications.**

B. Sc (Information Technology)		Semester – I	
Course Name: Fundamentals of Database Management Systems		Course Code: USIT103	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	75
	Internal	--	25

Course Objectives:
The objective of the course is to present an introduction to fundamentals of database management systems, with an emphasis on how to organize, maintain and retrieve - efficiently, and effectively - information from a DBMS.

Unit	Details	Lectures
I	Database system- concept and Architecture, Relational model and Relational database constraints. Relational Algebra.	12
II	Conceptual modelling and database design: Data modelling using the Entity Relationship model (ER).The enhanced entity relationship model. Relational database design by ER and EER model. Practical database design methodology and use of UML diagrams.	12
III	Database Design theory and normalization: Basics of functional dependencies and normalization for relational databases. Relational database design and further dependencies.	12
IV	Introduction to SQL , Complex queries, triggers, views, joining database tables and schema modification. Query Processing and optimization. File structure, hashing and indexing	12
V	Transaction management and concurrency control and recovery: Introduction to transaction processing concepts and theory. Concurrency control technique. Database recovery technique.	12

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Fundamentals of Database systems.	Ramez Elmasri, Shamkant B Navathe	Pearson.	6th Edition	
2.	Database Systems: Design implementation and management.	Carlos Coronel, Steven Morris, Peter Rob	Cengage Learning	9th Edition	2010

Course Outcomes:

Learners will be able to

1. Define and describe the fundamental elements of relational database management system.
2. To relate the basic concepts of relational data model, entity-relationship model, relational database design, relational algebra and SQL.
3. Design ER-models to represent simple database application scenarios.
4. Transform the ER-model to relational tables, populate relational database and formulate SQL queries on data.
5. Improve the database design by normalization.
6. Understand basic database storage structures and access techniques: file and page organizations, indexing methods and hashing.

B. Sc (Information Technology)		Semester – I	
Course Name: Fundamentals of Database Management Systems Practical		Course Code: USIT1P3	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2½	50
	Internal	--	--

Course Objectives:
- To introduce ER data model, database design and normalization.
- To Learn SQL basics for data definition and data manipulation.

1.	Draw E-R diagram and convert entities and relationships to relation table for a given scenario
a.	Bank
b.	College
2.	**Write relational algebra queries for a given set of relations**
3.	**Defining data**
a.	Using CREATE statement
b.	Using ALTER statement
c.	Using DROP statement
d.	Using TRUNCATE statement
e.	Using RENAME statement
4.	**Manipulating data**
a.	Using INSERT statement
b.	Using UPDATE statement
c.	Using DELETE statement
d.	Using SELECT statement
5.	**Creating and managing the tables**
a.	Creating table with contraints: NOTNULL, UNIQUE, PRIMARY KEY ,FOREIGN KEY
6.	**Restricting and sorting data**
a.	Using DISTINCT,IN, AS, SORT,LIKE,ISNULL, OR
b.	Using Group By, Having clause, Order By clause
7.	**Aggregate and Mathematical functions:**
a.	AVG,MIN,MAX,SUM,COUNT

b.	ABS,SQRT,ROUND,TRUNCATE,SIGN,POWER,MOD,FLOOR,CEIL
8.	**Views and Joins: For a given set of relation tables perform the following**
a.	Creating view
b.	Dropping view
c.	Selecting from a view
8.	**Database trigger**
a.	Using CREATE OR REPLACE TRIGGER
9.	**Index**
a.	Create index
b.	Drop index

Course Outcomes:

Learners will be able to:

1. **Design database schema for a given application and apply normalization.**
2. **Acquire skills in using SQL Commands for data Definition and data manipulation.**

B. Sc (Information Technology)			Semester – I	
Course Name: Computational Logic and Discrete Structures			**Course Code: USIT104**	
Periods per week (1 Period is 50 minutes)			**5**	
Credits			**2**	
			Hours	Marks
Evaluation System		Theory Examination	2	75
		Internal	--	25

Course Objectives:

- **Course will provide students with an overview of discrete mathematics.**
- **Students will learn about topics such as logic and proofs, sets and functions, recursion, graph theory, tress and other important discrete math concepts.**

Unit	Details	Lectures
I	**Set Theory** Introduction, Sets and Elements, Subsets, Venn Diagrams, Set Operations, Algebra of Sets, Duality, Finite Sets, Counting Principle, Classes of Sets, Power Sets, Partitions, Mathematical Induction **Relations** Introduction, Product Sets, Relations, Pictorial Representatives of Relations, Composition of Relations, Types of Relations, Closure Properties, Equivalence Relations, Partial Ordering Relations	12
II	**Functions and Algorithms** Introduction, Functions, One-to-One, Onto, and Invertible Functions, Mathematical Functions, Exponential and Logarithmic Functions, Sequences, Indexed Classes of Sets, Recursively Defined Functions, Cardinality, Algorithms and Functions, Complexity of Algorithms **Probability** Introduction, Sample Space and Events, Finite Probability Spaces, Conditional Probability, Independent Events, Independent Repeated Trials, Binomial Distribution, Random Variables, Chebyshev's Inequality, Law of Large Numbers	12
III	**Techniques of Counting** Introduction, Basic Counting Principles, Mathematical Functions, Permutations, Combinations, the Pigeonhole Principle, The Inclusion–Exclusion Principle, Tree Diagrams **Advanced Counting Techniques, Recursion** Introduction, Combinations with Repetitions, Ordered and Unordered Partitions, Inclusion–Exclusion Principle Revisited, Pigeonhole Principle Revisited, Recurrence Relations, Linear Recurrence Relations with Constant Coefficients, Solving Second-Order Homogeneous Linear Recurrence, Relations, Solving General Homogeneous Linear Recurrence Relations	12
IV	**Graph Theory** Introduction, Data Structures,,Graphs and Multigraphs, Subgraphs, Isomorphic and Homeomorphic Graphs, Paths, Connectivity, Traversable and Eulerian Graphs, Bridges of Königsberg, Labeled andWeighted Graphs, Complete, Regular, and Bipartite	12

	Graphs, Tree Graphs, Planar Graphs, Graph Colorings, Representing Graphs in Computer Memory, Graph Algorithms, Traveling-Salesman Problem, Solved Problems **Directed Graphs** Introduction, Directed Graphs, Basic Definitions, Rooted Trees, Sequential Representation of Directed Graphs, Warshall's Algorithm, Shortest Paths, Linked Representation of Directed Graphs, Graph Algorithms: Depth-First and Breadth-First Searches, Directed Cycle-Free Graphs, Topological Sort, Pruning Algorithm for Shortest Path	
V	**Binary Trees** Introduction, Binary Trees,, Complete and Extended Binary Trees, Representing Binary Trees in Memory, Traversing Binary Trees, Binary Search Trees, Priority Queues, Heaps, Path Lengths, Huffman's Algorithm, General (Ordered Rooted) Trees Revisited **Ordered Sets and Lattices** Introduction, Ordered Sets, Hasse Diagrams of Partially Ordered Sets, Consistent Enumeration, Supremum and Infimum, Isomorphic (Similar) Ordered Sets, Well-Ordered Sets, Lattices 346 Bounded Lattices, Distributive Lattices, Complements, Complemented Lattices	12

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Discrete Mathematics, Schaum's Outlines Series	Seymour Lipschutz, Marc Lipson	Tata MCGraw Hill	3rd	2007
2.	Discrete Mathematics with Applications	Sussana S. Epp	Cengage Learning	5th	2018
3.	Discrete Mathematics and its Applications	Kenneth H. Rosen	Tata MCGraw Hill	8th	2019
4.	Discrete mathematical structures	B Kolman RC Busby, S Ross	PHI		
5.	Discrete structures	Liu	Tata MCGraw Hill		

Course Outcomes:

Learners will be able to:

1. Use logical notation
2. Perform logical proofs
3. Apply recursive functions and solve recurrence relations
4. Use graphs and trees
5. Apply basic and advanced principles of counting
6. Define sets and Relations
7. Calculate discrete probabilities.

B. Sc (Information Technology)	Semester – I	
Course Name: Computational Logic and Discrete Structures Practical	Course Code: USIT1P4	
Periods per week (1 Period is 50 minutes)	3	
Credits	2	
	Hours	Marks
Evaluation System	Practical Examination	2½	50
	Internal	--	--

Course Objectives:

Course will make students understand different commands and functions of SCILAB. It will enable student to use these tools to compute solutions of various discrete mathematical structures.

1.	**Set Theory**
a.	Inclusion Exclusion principle.
b.	Power Sets
c.	Mathematical Induction
2.	**Functions and Algorithms**
a.	Recursively defined functions
b.	Cardinality
c.	Polynomial evaluation
d.	Greatest Common Divisor
3.	**Probability Theory 1**
a.	Sample space and events
b.	Finite probability spaces
c.	Equiprobable spaces
d.	Addition Principle
4.	**Probability Theory 2**
a.	Conditional Probability
b.	Multiplication theorem for conditional probability
c.	Independent events
d.	Repeated trials with two outcomes
5.	**Counting 1**
a.	Sum rule principle
b.	Product rule principle
c.	Factorial
d.	Binomial coefficients
6.	**Counting 2**
a.	Permutations
b.	Permutations with repetitions

c.	Combinations
d.	Combinations with repetitions
7.	**Counting 3**
a.	Ordered partitions
b.	Unordered partitions
8.	**Graph Theory**
a.	Paths and connectivity
b.	Minimum spanning tree
c.	Isomorphism
9.	**Directed Graphs**
a.	Adjacency matrix
b.	Path matrix
10	**Recurrence relations**
a.	Linear homogeneous recurrence relations with constant coefficients
b.	Solving linear homogeneous recurrence relations with constant coefficients
c.	Solving general homogeneous linear recurrence relations

Course Outcomes:

Learners will be able to:

1. **To find computational solution to various discrete mathematical structures.**

B. Sc (Information Technology)	Semester – I		
Course Name: Technical Communication Skills	Course Code: USIT105		
Periods per week (1 Period is 50 minutes)	5		
Credits	2		
		Hours	Marks
Evaluation System	Theory Examination	2	75
	Internal	--	25

Course Objectives:
- To recognize the importance of various types of communication in technical set up.
- To understand the dynamics in different forms of formal communication.
- To learn about active listening and the art of giving presentations and interviews.
- To learn the art of business writing and ethics in business communication across functional areas.
- To evaluate, analyze and interpret technical data.

Unit	Details	Lectures
I	**Fundamentals of Technical Communication** Introduction, The process of communication, Language as tool of communication, levels of communication, The flow of communication, Communication Networks, The importance of technical communication **Barriers to communication** Definition of Noise, classification of Barriers **Non-verbal Communication** Introduction, Definition, significance of nonverbal, forms of non -verbal communication, types of non-verbal communication	12
II	**The Seven Cs of Effective Communication**: Completeness, Conciseness, Consideration, Concreteness, Clarity, Courtesy, Correctness **Conversations** Introduction, Importance of Business conversion, Essential of Business conversion, Conversation Management **Meeting and conferences** Introduction, Purpose of Meeting, planning a meeting, Meeting Process, Leading effective meeting, Evaluating meeting, planning conference, teleconferencing **Group Discussion and team presentation** Introduction, Benefits of GD, Workplace GD guidelines, Functional and non functional roles in GD, Improving group performance, Assessment of group discussion ,Team presentation **Email communication** Introduction, Advantages of email, problems in email communication, Email etiquettes, Techniques of writing Effective Email	12
III	**Active Listening** Introduction, Type of listening, Traits of good listener, Active vs Passive listening, Implication of effective listening **Effective presentation Strategies** Introduction, Defining purpose, Analyzing audience and Locale, Organizing contents, preparing outline, Visual Aids, Understanding Nuances of delivery, Kinesics **Interview** Introduction, objectives, types of interview, job interviews	12
IV	**Business writing** Introduction, Importance of written Business, Five main strategies of writing business messages	12

	Business correspondence Business letter writing, common component of Business letter, Strategies for writing body of a letter, Types of Business letter, writing memos **Business reports and proposal** What is report? Steps in writing routine Business report, parts of report, corporate reports and Business proposals **Careers and Resume** Introduction to career building, resume format, traditional, electronic and video resumes, sending resume, follow up letters and online recruitment process	
V	**Communication across Functional areas** Financial communication, MIS **Ethics in Business Communication** Ethical communication, Values, ethics and communication, ethical dilemmas facing manager, strategic approaches to corporate ethics **Creating and Using Visual Aids** Object, Models, Handouts, Charts and Graphs, Text Visuals , Formatting Computer generated charts, graphs and visuals	12

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Technical communication : principles and practices	Meenakshi Raman & Sangeeta Sharma	Oxford Higher Education		
2.	Business Communication	Meenakshi Raman & Prakash Singh	Oxford- Higher Education	2nd edition	2006
3.	Effective Business Communication	Herta Murphy, Herbert Hildebrandt, Jane Thomas	Tata McGraw Hill	7th edition	2008
4.	Professional Communication	Aruna Koneru	McGraw Hill		2008
5.	Business and Professional Communication Plans, Processes and Performance	James R. DiSanza Nancy J..Legge	Pearson Education	4th Edition	
6.	Storytelling with data-a data visualization guide for business professionals	Cole Nussbaumer knaflic	Wiley		

Course Outcome:

Learners will be able to,

1. **Analyze, synthesize and utilize the process and strategies from delivery to solving communication problem.**
2. **Learn the communication methodologies at workplace and learning about importance of team collaboration.**
3. **Learn about different technical communication such as presentations and interviews.**
4. **Understand and apply the art of written communication in writing reports, proposals.**
5. **Ground rules of ethical communication and MIS.**
6. **Understand the functions of graphs, maps, charts.**

B. Sc (Information Technology)	Semester – I
Course Name: Technical Communication Skills Practical	**Course Code: USIT1P5**
Periods per week (1 Period is 50 minutes)	3
Credits	2

			Hours	Marks
Evaluation System		**Practical Examination**	2½	50
		Internal	--	--

Course Objectives:
- **To express thoughts feelings and ideas of learners by using features of MS Word.**
- **To articulate formal and informal reports.**
- **To analyze and interpret data and learn visualization of data.**
- **To learn effective tools of presentation.**

1.	Use of word processing tools for communication.
a.	Use of various tools like spell checker, header, footer etc.
b.	Make formal and informal letters, creating resume.
c.	Designing brochures and flyers using templates in word.
2.	Writing reports, minutes of meeting, action plan.
3.	Use of spreadsheet for data interpretation and data analysis.
4.	Basic use of what if analysis using excel.
5.	Visual Representation of data using excel – pie chart ,line chart, bar chart etc.
6.	Summarization of data using of pivot tables and chart in excel.
7.	Use of presentation tools like PowerPoint for communication and presentation skills.
8.a.	Basic communication covering the following topics:- Meeting people, Asking Questions and Design of questionnaire.
8.b.	Using netiquettes in online mode of communication using Zoom / Google Meet / MS-Teams etc.
9.	Use of Mail etiquette for writing effective mails.
10.a.	Use of Mail merge and its features.
10.b.	Creating profile using LinkedIn.

Course Outcome:

Learners will be able to:

1. **Use different forms of digital mediums for effective communication.**
2. **Create technical documents and format existing documents for effective communication.**
3. **Learn to use graphical tools for better visualization.**
4. **Create business presentation effectively.**
5. **Visualize the data from pictorial representations.**

SEMESTER II

B. Sc (Information Technology)	Semester – I
Course Name: Object Oriented Programming with C++	Course Code: USIT201
Periods per week (1 Period is 50 minutes)	5
Credits	2

			Hours	Marks
Evaluation System		Theory Examination	2	75
		Internal	--	25

Course Objectives:
Understand object oriented programming and advanced C++ concepts
- Be able to explain the difference between object oriented programming and procedural programming.
- Be able to program using more advanced C++ features such as composition of objects, operator overloads, dynamic memory allocation, inheritance and polymorphism, file I/O, exception handling, etc.
- Be able to build C++ classes using appropriate encapsulation and design principles
- Be able to apply object oriented or non-object oriented techniques to solve
- bigger computing problems

Unit	Details	Lectures
I	**INTRODUCTION OF OBJECT-ORIENTED DESIGN** : Introduction, Objects, Class and Instance, Polymorphism, Inheritance, Object-Oriented Analysis , Finding the Objects ,Conceptual Modeling Requirements Model , Analysis Model, The Design Model, The Implementation Model , Test Model, Object-Oriented Analysis and Design, The Evolution of Object Model, Object-Oriented Programming, Object-Oriented Design, Object-Oriented Analysis, Elements of Object Model ,The Role of OOAD in the Software Life Cycle, OOAD Methodologies, Grady Booch Approach, **STARTING WITH C++:** C++ Overview, C++ Character Set, C++ Tokens, Variables, Counting Tokens, Data Types, Qualifiers, Range of Data Types, Your First C++ Program, Structure of a C++ Program, Styles of, Writing C++ Programs, Programming Examples **FEATURES OF C++:** Introduction, Operators and Expressions, Declaring Constants, Type Conversion, Decision Making: An Introduction, Unconditional Branching Using Goto , Introduction to Looping **OPERATORS AND REFERENCES IN C++**: Introduction, Scope Resolution Operator, Reference Variables, The Bool Data Type, The Operator New and Delete, Malloc Vs. New ,Pointer Member Operators	12
II	**FUNCTION IN C++**: Introduction ,Function Declaration/Prototyping ,,The Main Function in C++,Recursion ,Call by Reference ,Call by Reference Vs Call by Address , Return by Reference ,,Inline Function ,Function Overloading ,Function with Default Arguments **CLASS AND OBJECTS IN C++** : Working with Class, Structure in C++ ,Accessing Private Data Passing and Returning Object ,Array of Object ,Friend Function ,Static Class Members ,Constant Member Function **WORKING WITH CONSTRUCTOR AND DESTRUCTOR:** Introduction, Constructor with Parameters, Implicit and Explicit Call to Constructor, Copy Constructor, Dynamic Initialization of Objects, Dynamic Constructor, Destructor **WORKING WITH OPERATOR OVERLOADING:** Introduction, Operator Overloading with Binary Operator Overloading Assignment (=) Operator, Overloading	12

	Unary Operators, Overloading Using Friend Function, Rules of Operator Overloading, Type Conversion	
III	**WORKING WITH INHERITANCE IN C++:** Introduction, Types of Inheritance, Public, Private and Protected Inheritance, Multiple Inheritance, Hierarchical Inheritance, Virtual Base Class, Constructor and Destructor in Inheritance, Containership **POINTERS TO OBJECTS AND VIRTUAL FUNCTIONS:** Pointer to Objects, The This Pointer, What is Binding in C++? , Virtual Functions ,Working of a Virtual Function ,Rules for Virtual Function ,Pure Virtual Function and Abstract Class ,Object Slicing ,Some Facts about Virtual Function ,Virtual Destructor **INPUT-OUTPUT AND MANIPULATORS IN C++:** Introduction, C++ Stream Classes, Unformatted Input/Output, Formatted Input /Output Operations, Manipulators	12
IV	**FILE HANDLING IN C++:** Introduction, File Streams, Opening and Closing a File, File Opening Modes Checking End of File, Random Access in File, Command Line Arguments, Working with Binary Mode Error Handling **TEMPLATE PROGRAMMING:** Introduction , Function Template , Class Template **EXCEPTION HANDLING IN C++ :** Introduction , Basics of Exception Handling , Exception Handling Mechanism , Programming Examples ,Exception Handling with Class Catching all Exceptions , Specifying Exception for a Function	12
V	**INTRODUCTION TO THE STANDARD TEMPLATE LIBRARY** : Introduction , Components of STL , Containers , Algorithms , Iterators ,Application of Container Classes Function Objects **MANIPULATING STRINGS** :Introduction , Creating (string) Objects , Manipulating String Objects , Relational Operations, String Characteristics , Accessing Characters in Strings , Comparing and Swapping **NEW FEATURES OF ANSI C++ STANDARD** : Introduction ,New Data Types , New Operators , Class Implementation , Namespace Scope , Operator Keywords , New Keywords , New Headers	12

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Object-oriented Programming C++ Simplified	Hari Mohan Pandey	University Science Press	1st Edition	2017
2.	Object Oriented Programming in C++	E Balagurusamy	Tata McGraw-Hill	5th Edition	2011
3.	Object-Oriented Programming in C++	Robert Lafore	Sams	4th Edition	2002
4.	Programming with ANSI C++	Bhushan Trivedi	Oxford University Press	2nd Edition	2012
5.	Demystified Object-Oriented Programming with C++	Dorothy R. Kirk	Packt Publishing Lt	1st Edition	2021
6.	C++ Programming: An Object-Oriented Approach	Behrouz A. Forouzan , Richard F. Gilberg	McGraw-Hill Education	1st edition	2020
7.	C++ How to Program	Paul Deitel, Harvey Deitel	Pearson Education	10th Edition	2017

Course Outcomes:

Learners will be able to,

1. Understand the concept of OOPs, feature of C++ language.
2. Understand and apply various types of Datatypes, Operators, Conversions while designing the program.
3. Understand and apply the concepts of Classes &Objects, friend function, constructors & destructors in program design.
4. Design & implement various forms of inheritance, String class, calling base class constructors.
5. Apply & Analyze operator overloading, runtime polymorphism, Generic Programming.
6. Analyze and explore various Stream classes, I/O operations and exception handling.

B. Sc (Information Technology)		Semester – II	
Course Name: Object Oriented Programming with C++ Practical		Course Code: USIT2P1	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2½	50
	Internal	--	--

Course Objectives:

- The student should be able to explain the important characteristics of the C++ programming language.
- The learner must be able to combine components of the C++ programming language to develop structured program.
- The student must demonstrate the skills essential to compile, debug, and test C++ programs correctly.

1.	
a.	Write a C++ program to create a simple calculator.
b.	Write a C++ program to convert seconds into hours, minutes and seconds.
c.	Write a C++ program to find the volume of a square, cone, and rectangle.
2.	
a.	Write a C++ program to find the greatest of three numbers.
b.	Write a C++ program to find the sum of even and odd n natural numbers
c.	Write a C++ program to generate all the prime numbers between 1 and n, where n is a value supplied by the user.
3.	
a.	Write a C++ program using classes and object Student to print name of the student, roll_no. Display the same.
b.	Write a C++ program for Structure bank employee to print name of the employee, account_no. & balance. Display the same also display the balance after withdraw and deposit
c.	Write a C++ Program to design a class having static member function named showcount() which has the property of displaying the number of objects created of the class.

d.	Write a Program to find Maximum out of Two Numbers using friend function. Note: Here one number is a member of one class and the other number is member of some other class.
e.	Write a C++ Program using copy constructor to copy data of an object to another object.
f.	Write a C++ Program to allocate memory dynamically for an object of a given class using class's constructor.
4.	
a.	Write a C++ program to design a class representing complex numbers and having the functionality of performing addition & multiplication of two complex numbers using operator overloading.
b.	Write a C++ program to overload new/delete operators in a class.
c.	Write a C++ program to access members of a STUDENT class using pointer to object members
	Write a C++ Program to generate Fibonacci Series by using Constructor to initialize the Data Members.
d.	Write a C++ Program to generate Fibonacci Series by using Constructor to initialize the Data Members.
e.	Write a C++ Program that illustrate single inheritance.
f.	Write a C++ Program that illustrate multipe inheritance.
g.	Write a C++ Program that illustrate multi level inheritance.
h.	Write a C++ Program that illustrate Hierarchical inheritance.
i.	Write a C++ Program illustrating how the constructors are implemented and the order in which they are called when the classes are inherited. Use three classes named alpha, beta, gamma such that alpha,beta are base class and gamma is derived class inheriting alpha & beta
5.	
a.	Write a C++ Program to design a stuent class representing student roll no. and a test class (derived class of student) representing the scores of the student in various subjects and sports class representing the score in sports. The sports and test class should be inherited by a result class having the functionality to add the scores and display the final result for a student.
6.	
a.	Write a C++ program to maintain the records of person with details (Name and Age) and find the eldest among them. The program must use this pointer to return the result.
7.	
a.	Write a C++ program illustrating the use of virtual functions in class.
b.	Write a C++ program to design a class representing the information regarding digital library (books, tape: book & tape should be separate classes having the base class as media). The class should have the functionality for adding new item, issuing, deposit etc. the program should use the runtime polymorphism.
8.	
a.	Write a C++ program to show conversion from string to int and vice-versa.
b.	Write a C++ program implementing basic operation of class ios i.e. setf, unsetf, precision etc.
c.	Write a C++ program to implement I/O operations on characters. I/O operations includes inputting a string, Calculating length of the string, Storing the String in a file, fetching the stored characters from it, etc.
d.	Write a C++ program to copy the contents of one file to another.

e.	Write a C++ program to perform read/write binary I/O operation on a file (i.e. write the object of a structure/class to file).
9.	
a.	Write a C++ program to implement the exception handling with multiple catch statements.
b.	Write a C++ program to implement the exception handling with rethrowing in Exception.
10.	
a.	Write a C++ Program to create Simple calculator using Class template.
b.	Write a C++ Program to get maximum of two number using Class template.

Course Outcomes:

Leaners will able to,

- **Utilize C++ characteristics in software design and development.**
- **Explain object-oriented techniques and explain how C++ supports them.**
- **Employ C++ to demonstrate practical skill developing object-oriented solutions.**
- **Examine a problem statements and design and develop object-oriented software using good coding practices and procedures.**
- **In object-oriented design, use common software patterns and recognize their relevance in other software development contexts.**

B. Sc (Information Technology)		Semester – II	
Course Name: Fundamentals of Micro Processor and Microcontrollers		Course Code: USIT202	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	75
	Internal	--	25

Course Objectives: 1) To understand the basic concept of Micro Computer Systems
2) To develop background knowledge in 8085 Microprocessor
3) To write Assembly language Programs of 8085
4) To understand the peripheral devices and interfacing to 8051 Micro Controller and design aspects of Micro Controller

Unit	Details	Lectures
I	**Microprocessor, microcomputers, and Assembly Language:** Microprocessor,Microprocessor Instruction Set and Computer Languages, From Large Computers to Single-Chip Microcontrollers, Applications. **Microprocessor Architecture and Microcomputer System:** Microprocessor Architecture and its operation's, Memory, I/O Devices, Microcomputer System, Logic Devices and Interfacing, Microprocessor-Based System Application. **8085 Microprocessor Architecture and Memory Interface:** Introduction, 8085 Microprocessor unit, 8085-Based Microcomputer, Memory Interfacing,Interfacing the8085 Memory Segment.	12
II	**Interfacing of I/O Devices** Basic Interfacing concepts, Interfacing Output Displays, Interfacing Input Devices, Memory Mapped I/O, Testing and Troubleshooting I/O Interfacing Circuits. **Introduction to 8085 Assembly Language Programming:** The 8085 Programming Model, Instruction Classification, Instruction, Data and Storage, Writing assembling and Execution of a simple program, Overview of 8085 Instruction Set, Writing and Assembling Program. **Introduction to 8085 Instructions:** Data Transfer Operations, Arithmetic Operations, Logic Operation, Branch Operation, Writing Assembly Languages Programs, Debugging a Program.	12
III	**Programming Techniques With Additional Instructions:** ProgrammingTechniques: Looping, Counting and Indexing, Additional Data Transfer and 16-Bit Arithmetic Instructions, Arithmetic Instruction Related to Memory, Logic Operations: Rotate, Logics Operations: Compare, Dynamic Debugging. **Counters and Time Delays:** Counters and Time Delays, Illustrative Program: Hexadecimal Counter, Illustrative Program: zero-to-nine (Modulo Ten) Counter, Generating Pulse Waveforms, Debugging Counter and Time-Delay Programs. **Stacks and Sub-Routines:** Stack, Subroutine, Restart, Conditional Call, Return Instructions, Advanced Subroutine concepts. **Interrupts:** The 8085 Interrupt, 8085 Vectored and Non vectored Interrupts, Restart as S/W Instructions.	12

IV	**Micro Controllers:** Embedded Systems and general purpose computer systems, history, classifications, applications and purpose of embedded systems. **Embedded Hardware:** Memory map, i/o map, interrupt map, processor family, external peripherals, memory – RAM , ROM, types of RAM and ROM, memory testing, CRC ,Flash memory. **Peripherals:** Control and Status Registers, Device Driver, Timer watch Timer **The 8051 Microcontrollers:** Microcontrollers and Embedded processors, Overview of 8051 family.8051 Microcontroller hardware, Input/output pins, Ports, and Circuits, External Memory. **8051 Programming in C:** Data Types and time delay in 8051 C, I/O Programming, Logic operations, Data conversion Programs.	12
V	**Designing Embedded System with 8051 Microcontroller:**Factors to be considered in selecting a controller, why 8051 Microcontroller, Designing with 8051. **Programming embedded systems:** structure of embedded program, infinite loop, compiling, linking and debugging. **Design and Development:** Embedded system, development Environment – IDE, types of file generated on cross compilation, Embedded Product Development cycle and Trends in embedded Industry	12

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
1.	Microprocessors Architecture, Programming and Applications with the 8085.	Ramesh Gaonkar	PENRAM	5th	2012
2.	8080A/8085 Assembly Language Programming	Lance A. Leventhel	Osborne		1978
3	Embedded Systems	Rajkamal	Tata Mcgraw-Hill		
4	Introduction to embedded systems	Shibu K V	Tata Mcgraw-Hill	1st	2012

Course Outcomes:

Learners will be able to,

1. **Understand the basic concepts of Micro Computer Systems**
2. **Understand the architecture and hardware aspects of 8085**
3. **Write assembly language programs in 8085**
4. **Design elementary aspects of Micro Controller based systems**
5. **Interfacing peripherals using Micro Controller**

B. Sc (Information Technology)		Semester – II	
Course Name: Fundamentals of Micro Processor and Microcontrollers Practical		Course Code: USIT2P2	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	Marks
Evaluation System	Practical Examination	2½	50
	Internal	--	--

Course Objectives: The course aims to learn
1. Operations related to single & Multiple memory locations
2. Simple assembly language programs
3. How to perform register operations, packing and unpacking
4. Embedding computer using 8051 microcontrollers
5. Interfacing I/O Ports
6. Implement flashmagic in microcontrollers

List of Practical	
1.	**Perform the following Operations related to memory locations.**
a.	Store the data byte 32H into memory location 4000H.
b.	Exchange the contents of memory locations 2000H and 4000H
2.	**Simple assembly language programs.**
a.	Subtract two 8-bit numbers.
b.	Subtract the 16-bit number in memory locations 4002H and 4003H from the 16-bit number in memory locations 4000H and 4001H. The most significant eight bits of the two numbers are in memory locations 4001H and 4003H. Store the result in memory locations 4004H and 4005H with the most significant byte in memory location 4005H.
c.	Find the 1's complement of the number stored at memory location 4400H and store the complemented number at memory location 4300H.
d.	Find the 2's complement of the number stored at memory location 4200H and store the complemented number at memory location 4300H.
3.	**Packing and unpacking operations.**
a.	Pack the two unpacked BCD numbers stored in memory locations 4200H and 4201H and store result in memory location 4300H. Assume the least significant digit is stored at 4200H.
b.	Two digit BCD number is stored in memory location 4200H. Unpack the BCD number and store the two digits in memory locations 4300H and 4301H such that memory location 4300H will have lower BCD digit.
4.	**Register Operations.**
a.	Write a program to shift an eight bit data four bits right. Assume that data is in register C.
b.	Program to shift a 16-bit data 1 bit left. Assume data is in the HL register pair

c.	Write a set of instructions to alter the contents of flag register in 8085.	
d.	Write a program to count number of 1's in the contents of D register and store the count in the B register.	
5.	**Multiple memory locations.**	
a.	Calculate the sum of series of numbers. The length of the series is in memory location 4200H and the series begins from memory location 4201H. a. Consider the sum to be 8 bit number. So, ignore carries. Store the sum at memory location 4300H. b. Consider the sum to be 16 bit number. Store the sum at memory locations 4300H and 4301H	
b.	Multiply two 8-bit numbers stored in memory locations 2200H and 2201H by repetitive addition and store the result in memory locations 2300H and 2301H.	
c.	Divide 16 bit number stored in memory locations 2200H and 2201H by the 8 bit number stored at memory location 2202H. Store the quotient in memory locations 2300H and 2301H and remainder in memory locations 2302H and 2303H.	
6.	**Calculations with respect to memory locations.**	
a.	Write a program to sort given 10 numbers from memory location 2200H in the ascending order.	
b.	Calculate the sum of series of even numbers from the list of numbers. The length of the list is in memory location 2200H and the series itself begins from memory location 2201H. Assume the sum to be 8 bit number so you can ignore carries and store the sum at memory location 2Sample problem:	
7.	**Assembly programs on memory locations**	
a.	A list of 50 numbers is stored in memory, starting at 6000H. Find number of negative, zero and positive numbers from this list and store these results in memory locations 7000H, 7001H, and 7002H respectively	
b.	Write an assembly language program to generate fibonacci number.	
c.	Program to calculate the factorial of a number between 0 to 8.	
8.	Design and develop a reprogrammable embedded computer using 8051 microcontrollers and to show the following aspects. a. Programming b. Execution c. Debugging	
9 a	Configure timer control registers of 8051 and develop a program to generate given time delay.	
b	Port I / O: Use one of the four ports of 8051 for O/P interfaced to eight LED's. Simulate binary counter (8 bit) on LED's	
c	To interface 8 LEDs at Input-output port and create different patterns.	
d	To demonstrate timer working in timer mode and blink LED without using any	

	loop delay routine.
10.	**Using FlashMagic**
a.	To demonstrate the procedure for flash programming for reprogrammable embedded system board using Flash Magic
b.	To demonstrate the procedure and connections for multiple controllers programming of same type of controller with same source code in one go, using flash magic.

Course Outcomes:

Learners will be able to,

1. Apply concepts of 8085 to single & Multiple Memory Locations
2. Apply concepts of micro-processor register operations
3. Can implement assembly language programs
4. Use of Shift registers 8 & 16 bits
5. Apply the knowledge of Flash Magic in embedded Controllers
6. Learns to simulate and configure different timer controls

B. Sc (Information Technology)		Semester – II	
Course Name: Web Applications Development		Course Code: USIT203	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	75
	Internal	--	25

Course Objectives:
- Understand basic concepts of Internet and World Wide Web.
- Comprehend different HTML elements that can be used to develop static web pages.
- Become familiar with concept of stylesheets and various CSS effects.
- Peruse JavaScript as a tool to add dynamism to static HTML pages.
- Explore how server-side script works on the web.
- Learn how PHP can be connected to a database to store and retrieve data.

Unit	Details	Lectures
I	**Internet and the World Wide Web**: What is Internet? Applications of Internet, E-mail, Telnet, FTP, E-commerce and E-business. Internet Service Providers, Domain Name Server, Internet Address, World Wide Web (WWW): World Wide Web and its Evolution, Uniform Resource Locator (URL), Browsers, Common Features of Browsers, Search Engine, Web Server, HTTP Protocol. **HTML5**: Introduction, Formatting Text by using Tags, Using Lists, Creating Hyperlinks and Bookmarks, Defining Metadata about an HTML Document, Redirecting to another URL. **CSS**: Implementing Styles using CSS – Stylesheets, Formatting Text and Links using CSS, CSS Selectors, Changing Background, Adding Border, Margin and Padding, Setting Dimensions, Using Inline Container to mark up a part of a text.	12
II	**HTML Page Layout**: Using Layout Elements, Semantic Elements, Creating, Positioning and Formatting Divisions, Floating Divisions next to each other, Responsive Web Design, Inline Frames. **HTML Media, Tables and Forms**: Embedding Images, Creating Client-side and Server-side Image Map, adding Favicon, Embedding audio and video on web page. Creating Simple Table, Table Dimension, Merging Table Cells, Formatting Tables: Applying Borders, Background and Foreground fills, Changing Cell Padding, Spacing and Alignment Collecting user input with HTML Forms, Additional Input Types in HTML5.	12
III	**JavaScript**: Introduction, Difference between Client-side and Server-side Scripting, JavaScript Variables and Constants, Data Types, Statements, Comments, Functions, Variable Scope, Hoisting, Strict Mode, JavaScript Objects, Dialog Boxes, void Keyword	12

	Operators: Arithmetic Operators, Assignment Operators, Comparison Operators, Logical Operators, Bitwise Operators **Statements**: Conditional Statements – if else, switch, Loops – while, do while, for, for in, for of, Loop Control – break, continue, labels **JavaScript Objects**: User-defined Objects, with Keyword, Native Objects – Array, String, Date, Math, Number, RegExp **DOM**: Introduction, DOM Properties and Methods. **Browser BOM**: Moving back and forward with History, Cookies **Events and Event Handlers**: HTML Events, DOM Events, DOM Event Listener, onAbort, onBlur, onChange, onClick, onDblClick, onError, onFocus, onKeyDown, onKeyPress, onKeyUp, onLoad, onMouseDown, onMouseMove, onMouseOut, onMouseOver, onMouseUp, onReset, onResize, onSelect, onSubmit, onUnload		
IV	**PHP**: Introduction, Server-side Scripting, PHP Syntax and Comments, Variables and Constants, Data Types, Control Structures, Looping, Loop Termination, Functions, PHP Form Handling, PHP Form Validation, Superglobals, PHP Arrays, PHP Strings, PHP RegEx, PHP Numbers, PHP Math, Basic PHP Errors	12	
V	**Advanced PHP**: PHP Date and Time, PHP Include, PHP Cookies, PHP Sessions, Validating and Sanitizing Data with PHP Filters **PHP and MySQL** Why PHP and MySQL? Connect to MySQL, Creating Database and Tables, Inserting Single and Multiple Rows, Retrieving Last ID, MySQL Prepared, Selecting Data, Updating Data, Deleting Data, Limiting Data.	12	

Course Outcomes:
- **Analyze working of Internet.**
- **Gain an insight into designing web pages.**
- **Use different ways of styling web pages using CSS.**
- **Implement basic and complex functionalities of JavaScript in a web page.**
- **Employ PHP Scripts to execute dynamic tasks in a web page.**
- **Perform various database tasks using PHP.**

Books and References:					
Sr. No.	**Title**	**Author/s**	**Publisher**	**Edition**	**Year**
	The Complete Reference HTML & CSS	Thomas A. Powell	McGrawHill	5th	-
	Step by Step HTML5	Faithe Wempen	Microsoft Press		2011
	Learning PHP, MySQL, JavaScript, CSS & HTML5	Robin Nixon	O'Reilly	3rd	2018
	Learning Web Design A Beginner's Guide to Html, CSS, JavaScript, And Web Graphics	Jennifer Niederst Robbins	O'Reilly	5th	2018

| | The Complete Reference JavaScript | Thomas A. Powell & Fritz Schneider | McGrawHill | 3rd | 2012 |
| | PHP & MySQL Novice to Ninja | Tom Butler | SPD | 7th | 2022 |

B. Sc (Information Technology)		**Semester – II**	
Course Name: Web Application Development Practical		**Course Code: USIT2P3**	
Periods per week (1 Period is 50 minutes)		3	
Credits		2	
		Hours	**Marks**
Evaluation System	**Practical Examination**	2½	50
	Internal	--	--

Course Objectives:
- **Understand how to effectively implement HTML.**
- **Write CSS effectively to create well organized, styled web pages.**
- **Add versatility to a web page with client-side scripting.**
- **Deploy a local web server and run a simple web application.**
- **Read and process data in MySQL using PHP.**

List of Practical:	
1.	**Use of Basic Tags**
a.	Design a web page using different text formatting tags.
b.	Design a web page with links to different pages and allow navigation between web pages.
c.	Design a web page that automatically redirects the user to another page.
2.	**Use of CSS**
a.	Design a web page demonstrating different stylesheet types.
b.	Design a web page demonstrating grouping selectors.
3.	**Layout and Media**
a.	Design a web page demonstrating different semantics.
b.	Design a web page embedding image, audio and video.
c.	Design a web page with Imagemaps.
4.	**Tables and Forms**
a.	Design a web page with different tables.
b.	Design a web page with a form that uses all types of controls.

5.	**JavaScript**
a.	Using JavaScript, design a web page to accept a number from the user and print its Factorial.
b.	Using JavaScript, a web page that prints Fibonacci series/any given series.
c.	Write a JavaScript program to display all the prime numbers between 1 and 100.
d.	Write a JavaScript program to accept a number from the user and display the sum of its digits.
6.	**JavaScript Objects**
a.	Using JavaScript, design a web page demonstrating different native objects of JavaScript.
b.	Write a program in JavaScript to accept a sentence from the user and display the number of words in it. (Do not use split () function).
7.	**JavaScript Events**
a.	Write a JavaScript program to design simple calculator.
b.	Design a form and validate all the controls placed on the form using JavaScript.
8.	**Basic PHP**
a.	Write a PHP code to find the greater of 2 numbers. Accept the no. from the user.
b.	Write a PHP Program to accept a number from the user and print it factorial.
c.	Write a PHP program to accept a number from the user and print whether it is prime or not.
d.	Write a PHP program to display the following Binary Pyramid: 1 0 1 1 0 1 0 1 0 1 1 0 1 0 1
e.	Write a PHP program to demonstrate different string functions.
f.	Write a PHP program to demonstrate different array functions.
9.	**Advanced PHP**
a.	Write a PHP program to demonstrate use of sessions and cookies.
b.	Write a PHP program to demonstrate use of filters.
10.	**PHP and MySQL**
a.	Write a PHP program to create: Create a database College

	Create a table Department (Dname, Dno, Number_of_faculty)
b.	Write a PHP program to create a database named "College". Create a table named "Student" with following fields (sno, sname, percentage). Insert 3 records of your choice. Display the names of the students whose percentage is between 35 to 75 in a tabular format.

c.	Write a PHP program to
	    Update rows in a table
	    Delete rows from a table

d.	Design a PHP page for authenticating a user

Course Outcomes:

Learners will be able to,

1. **Design static web pages using Hyper Text Markup Language (HTML).**
2. **Enhance the look of web pages by implementing CSS.**
3. **Collect information from the user with HTML Forms.**
4. **Design interactive webpages using client-side script (JavaScript).**
5. **Implement Document Object Model and events in web pages using JavaScript.**
6. **Write and deploy basic PHP code to simplify web development.**
7. **Store and retrieve data from a server using PHP.**

B. Sc (Information Technology)		Semester – I	
Course Name: Numerical Methods		Course Code: USIT204	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	75
	Internal	--	25

Course Objectives:
Course will enhance the problem solving skills of students using extremely powerful numerical methods.

Unit	Details	Lectures
I	**Mathematical Modeling and Engineering Problem Solving:** A Simple Mathematical Model, Conservation Laws and Engineering Problems **Approximations and Round-Off Errors:** Significant Figures, Accuracy and Precision, Error Definitions, Round-Off Errors **Truncation Errors and the Taylor Series:** The Taylor Series, Error Propagation, Total Numerical Errors, Formulation Errors and Data Uncertainty	12
II	**Solutions of Algebraic and Transcendental Equations:** The Bisection Method, The Newton-Raphson Method, The Regula-falsi method, The Secant Method. **Interpolation:** Forward Difference, Backward Difference, Newton's Forward Difference Interpolation, Newton's Backward Difference Interpolation, Lagrange's Interpolation	12
III	**Solution of simultaneous algebraic equations (linear) using iterative methods:** Gauss-Jordan Method, Gauss-Seidel Method. **Numerical differentiation and Integration:** Numberical differentiation, Numerical integration using Trapezoidal Rule, Simpson's 1/3rd and 3/8th rules.	12
IV	**Numerical solution of 1st and 2nd order differential equations:** Taylor series, Euler's Method, Modified Euler's Method, Runge-Kutta Method for 1st and 2nd Order Differential Equations. **Least-Squares Regression:** Linear Regression, Polynomial Regression, Multiple Linear Regression, General Linear Least Squares, Nonlinear Regression	12
V	**Linear Programming:** Linear optimization problem, Formulation and Graphical solution, Basic solution and Feasible solution. **Numerical Solutions of Partial Differential Equations:** Classification of Partial Differential Equations of the second order, Difference equation corresponding to Laplace Equation, Liebmann's Iteration Process, Bender-Schmidt's Difference Equation corresponding to the parabolic equation, Crank Nicolson's difference equations corresponding to the parabolic equation, Difference equation corresponding to the Hyperbolic equation..	12

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year

1.	Introductory Methods of Numerical Methods	S. S. Sastry	PHI	5th	2012
2.	Numerical Methods for Engineers	Steven C. Chapra, Raymond P. Canale	Tata Mc Graw Hill	6th	2010
3.	Numerical Analysis	Richard L. Burden, J. Douglas Faires	Cengage Learning	9th	2011
4.	Numerical Methods	T Veerarajan T Ramachandran	Tata Mc Graw Hill	7th	2011

Course Outcomes:

Learners will be able to,

1. Understand numerical techniques to find the roots of non-linear equations and solution of system of linear equations.
2. Understand the difference operators and the use of interpolation.
3. Understand numerical differentiation and integration and numerical solutions of ordinary and partial differential equations.

B. Sc (Information Technology)	Semester – I	
Course Name: Numerical Methods Practical	**Course Code: USIT2P4**	
Periods per week (1 Period is 50 minutes)	3	
Credits	2	
	Hours	**Marks**
Evaluation System	**Practical Examination**	2½	50
	Internal	--	--

Course Objectives:

Course will provide different tools to find solutions to various numerical techniques

List of Practical: To be implemented using SCILAB	
1.	**Iterative Calculation**
a.	Program for iterative calculation
b.	Program to calculate the roots of a quadratic equation using the formula
c.	Program to evaluate e^x using infinite series
2.	**Solution of algebraic and transcendental equations:**

a.	Program to solve algebraic and transcendental equation by bisection method.
b.	Program to solve algebraic and transcendental equation by false position method.
c.	Program to solve algebraic and transcendental equation by Secant method.
d.	Program to solve algebraic and transcendental equation by Newton Raphson method.
3.	**Interpolation**
a.	Program for Newton's forward interpolation.
b.	Program for Newton's backward interpolation.
c.	Program for Lagrange's interpolation.
4.	**Solving linear system of equations by iterative methods**
a.	Program for solving linear system of equations using Gauss Jordan method.
b.	Program for solving linear system of equations using Gauss Seidel method.
5.	**Numerical Differentiation**
a.	Program to obtain derivatives numerically.
6.	**Numerical Integration**
a.	Program for numerical integration using Trapezoidal rule.
b.	Program for numerical integration using Simpson's 1/3rd rule.
c.	Program for numerical integration using Simpson's 3/8th rule.
7.	**Solution of differential equations**
a.	Program to solve differential equation using Euler's method.
b.	Program to solve differential equation using modified Euler's method.
c.	Program to solve differential equation using Runge-kutta 2nd order and 4th order methods.
8.	**Regression 1**
a.	Program for Linear regression.
b.	Program for Polynomial Regression.
9.	**Regression 2**

a.	Program for multiple linear regression.
b.	Program for non-linear regression.
10.	Numerical solution of partial differential equations
a.	Program to find solution of Laplace's equation.

Course Outcomes:

Learners will be able to,

1. **Find fast and accurate solution to simple and complex numerical problems using these programs.**

B. Sc (Information Technology)		Semester – II	
Course Name: Green IT		Course Code: USIT205	
Periods per week (1 Period is 50 minutes)		5	
Credits		2	
		Hours	Marks
Evaluation System	Theory Examination	2	75
	Internal	--	25

Course Objectives:
- To understand the concept of Green Technology.
- To learn Green IT regulating Green IT and different standards.
- To understand the concept of minimizing power utilization in technology.
- To know about Green PCs, Green notebooks and servers and Green data centers.
- To know how the way of work is changing and understand implementation of Paperless work.
- To know the concept of Recycling.
- To understand Metrics for Green IT.

Unit	Details	Lectures
I	**Overview to Green IT:** Problems: Toxins, Power Consumption, Equipment Disposal, Company's Carbon Footprint: Measuring, Details, reasons to bother, Plan for the Future, Cost Savings: Hardware, Power. **Regulating Green IT: Laws, Standards and Protocols** Introduction, The Regulatory Environment and IT Manufacturers RoHS, REACh, WEEE, Legislating for GHG Emissions and Energy Use of IT Equipment.Nonregulatory Government Initiatives, Industry Associations and Standards Bodies, Green Building Standards, Green Data Centres , Social Movements and Greenpeace.	12
II	**Minimizing Power Usage:** Power Problems, Monitoring Power Usage, Servers, Low-CostOptions, Reducing Power Use, Data De-Duplication, Virtualization, Management, Bigger Drives, Involving the Utility Company, LowPower Computers, PCs, Linux, Components, Servers, ComputerSettings, Storage, Monitors, Power Supplies, Wireless Devices, Software. **Cooling:** Cooling Costs, Power Cost, Causes of Cost, Calculating CoolingNeeds, Reducing Cooling Costs, Economizers, On-Demand Cooling, HP's Solution, Optimizing Airflow, Hot Aisle/Cold Aisle, Raised, Floors, Cable Management, Vapour Seal, Prevent Recirculation of Equipment Exhaust, Supply Air Directly to Heat Sources, Fans, Humidity, Adding Cooling, Fluid Considerations, System Design, Datacentre Design, Centralized Control, Design for Your Needs, Put Everything Together.	12
III	**Greening IT:** Green PCs, Notebooks and Servers, Green Data Centres , Green Cloud Computing, Green Data Storage , Green Software, Green Networking and Communications. **Changing the Way of Work:** Old Behaviours, starting at the Top, Process Reengineering with Green in Mind, Analysing the Global Impact of Local Actions, Steps: Water, Recycling, Energy, Pollutants, Teleworkers and Outsourcing, Telecommuting, Outsourcing, how to Outsource. **Going Paperless:** Paper Problems, The Environment, Costs: Paper and Office, Practicality, Storage, Destruction, Going Paperless, Organizational Realities, Changing Over, Paperless Billing, Handheld Computers vs. the Clipboard, Unified Communications, Intranets, What to Include, Building an Intranet, Microsoft Office SharePoint Server 2007, Electronic Data Interchange (EDI), Nuts and Bolts, Value Added Networks, Advantages, Obstacles.	12

IV	**Recycling:** Means of Disposal, Recycling, Refurbishing, Make the Decision, Life Cycle, from beginning to end, Life, Cost, Green Design, Recycling Companies, Finding the Best One, Checklist, Certifications, Hard Drive Recycling, Consequences, cleaning a Hard Drive, Pros and cons of each method, CDs and DVDs, good and bad about CD and DVDs disposal, Change the mind-set, David vs. America Online. **Hardware Considerations:** Certification Programs, EPEAT, RoHS, Energy Star, Computers, Monitors, Printers, Scanners, All-in-Ones, Thin Clients, Servers, Blade Servers, Consolidation, Products, Hardware Considerations, Planned Obsolescence, Packaging, Toxins, Other Factors, Remote Desktop, Using Remote Desktop, Establishing a Connection.	12
V	**Greening Your Information Systems:** Initial Improvement Calculations, Selecting Metrics, Tracking Progress, Change Business Processes, Customer Interaction, Paper Reduction, Green Supply Chain, Improve Technology Infrastructure, Reduce PCs and Servers, Shared Services, Hardware Costs, Cooling. **Staying Green:** Organizational Check-ups, Chief Green Officer, Evolution, Sell the CEO, SMART Goals, Equipment Check-ups, Gather Data, Tracking the data, Baseline Data, Benchmarking, Analyse Data, Conduct Audits, Certifications, Benefits, Realities, Helpful Organizations.	12

Books and References:					
Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Green IT	Toby Velte, Anthony Velte, Robert Elsenpeter	McGraw Hill		2008
2.	Harnessing Green IT: Principles and Practices	San Murugesan, G. R. Ganadharan,	Wiley & IEEE.		
3.	Green Data Center: Steps for the Journey	Alvin Galea, Michael Schaefer, Mike Ebbers	Shroff Publishers and Distributers		2011
4.	Green IT	Deepak Shikarpur	Vishwkarma Publications,		2014
5.	Green Computing Tools and Techniques for Saving Energy, Money and Resources	Bud E. Smith	CRC Press		2014
	Green Computing and Green IT Best Practice	Jason Harris	Emereo		

Course Outcomes:

Learners will be able to,

- **Understand the concept of Green IT and problems related to it.**
- **Know different standards for Green IT.**
- **Understand the how power usage can be minimized in Technology.**
- **Learn about how the way of work is changing.**
- **Understand the concept of recycling.**
- **Know how information system can stay Green Information system.**

B. Sc (Information Technology)			Semester – II	
Course Name: Practical's in PL/SQL			Course Code: USIT2P5	
Periods per week (1 Period is 50 minutes)			3	
Credits			2	
			Hours	Marks
Evaluation System		Practical Examination	2½	50
		Internal	--	--

Course Objectives:
- To understand the basics of PL/SQL.
- To understand control and conditional statement in PL/SQL.
- To understand working of sequences and cursor in PL/SQL.
- To understand concept of stored procedure and functions.
- To understand triggers and packages in PL/SQL.
- To understand the concept of Exception handling.

List of Practical:	
1.	**PL/SQL Basics**
a.	Use of variables.
b.	Write executable statement.
c.	Interacting with Oracle Server.
d.	Create anonymous PL/SQL block
2.	**Control Structure in PL/SQL .**
a.	Using while loop
b.	Do loop
c.	For loop
d.	Use of GOTO statement
3.	**Create conditional statement using PL/SQL**
a.	Using if statement
b.	Using if else statement
c.	Using elsif ladder
d.	Using case expression.
4.	**Creation of Sequence in PL/SQL**
5.	**Create cursor in PL/SQL**
a.	Implicit cursor
b.	Explicit

c.	Parameterized cursor
d.	Cursor for loop
6.	**Creation of Procedures in PL/SQL**
7.	**Functions in PL/SQL**
a.	Compute and returns the maximum value
b.	Compute factorial of given number.
8.	**Creation of Trigger**
a.	Create Row level trigger
b.	Create Statement level trigger
c.	Create instead of trigger
9.	**Handling exceptions**
a.	Creation of user defined exception
b.	Creation of system defined exception.
10.	**Creation of Package in PL/SQL**

Course Outcomes:

Learner will be able to:

- **Understand the basics of PL/SQL.**
- **Use of the control and conditional statement in PL/SQL.**
- **Apply sequences and cursor in PL/SQL.**
- **Know the concept of stored procedure and functions**
- **Create the triggers and packages in PL/SQL.**
- **Implement the concept of Exception handling.**